TOWARD BETTER SPEAKER EMBEDDINGS: AUTOMATED COLLECTION OF SPEECH SAMPLES FROM UNKNOWN DISTINCT SPEAKERS

*Minh Pham, Zeqian Li, and Jacob Whitehill**

Department of Computer Science, Worcester Polytechnic Institute (WPI)

ABSTRACT

The accuracy of speaker verification and diarization models depends on the quality of the speaker embeddings used to separate audio samples from different speakers. With the goal of training better embedding models, we devise an automatic pipeline for large-scale collection of speech samples from unique speakers that is significantly more automated than previous approaches. With this pipeline, we collect and publish the BookTubeSpeech dataset, containing 8,450 YouTube videos (7.74 min per video on average) that each contains a single unique speaker. Using this dataset combined with VoxCeleb2, we show a substantial improvement in the quality of embeddings when tested on LibriSpeech compared to a model trained on only VoxCeleb2.

***Index Terms*—** speaker embeddings, speech dataset, speaker verification, speaker diarization

1. INTRODUCTION

Two fundamental problems in automatic speech analysis are *speaker verification* and *speaker diarization*. The former is about testing whether a new audio recording belongs to a known speaker in a dataset. The latter is about estimating automatically *who is speaking when* from an audio recording in an unsupervised fashion. State-of-the-art approaches to both problems are based on *speaker embeddings*, which represent a speaker's identity so as to be independent of the speech content, emotion, background noise, and other factors. Over the past 10 years, researchers have devised a variety of embedding methods with increasing accuracy; examples include i-vectors [1] based on classical linear subspace methods, and d-vectors [2] and x-vectors [3], both of which are based on deep neural networks.

In speaker diarization and verification tasks, the accuracy of the final system depends to significant extent on how accurately the embedding function can separate different speakers. One of the main challenges to improving the embedding model is to collect a large dataset of multiple utterances from many thousands of *unique* speakers. The largest publicly available dataset of which we are aware is VoxCeleb2 [4], which contains 6,112 speakers from celebrities. This dataset is highly valuable and constitutes a significant research contribution. However, it is not obvious how the method used to collect VoxCeleb2 could be scaled because it was based on pre-selecting a set of Person of Interest (POI; see Related Work below). Large companies such as Google and Amazon presumably have their own massive datasets of speakers, but these are not publicly available.

In this paper, we present a scalable data collection pipeline that requires minimal human labor. Because it requires no pre-selection of POIs and instead is based on broad keyword searches it can scale to very large dataset sizes. Using this pipeline, we collect and release for academic use a new speech dataset called BookTubeSpeech. We also conduct several experiments on speaker verification to show the utility of this dataset for training new speaker embedding models.

2. RELATED WORK

The 3 main databases for speaker verification and diarization are VoxCeleb1 [5], VoxCeleb2 [4], and LibriSpeech [6]. **VoxCeleb 1 & 2**: These datasets were collected using the following procedure: First, a set of Persons of Interest (POI) is selected by hand. As indicated by the dataset's name, these largely consist of celebrities for which many speech recordings are available online. For each POI, a set of videos is downloaded using a web-based keyword search. Face detection, face identification, and visual speech detection are then used to verify that the person speaking at each moment is in fact the desired POI. Using this approach, VoxCeleb1 and VoxCeleb2 were created which contain 1,251 and 6,112 unique speakers, respectively. For each POI, multiple speech recordings are available. **LibriSpeech**: While intended created for speech recognition rather than verification or diarization, LibriSpeech does include labels of speaker identities and is thus useful for speaker embeddings. It consists of 1,000 hours from over 9,000 speakers reading audiobooks out loud. The speaker identities were extracted from the audiobook metadata, which were manually annotated.

Compared to the data pipelines to collect VoxCeleb2 and LibriSpeech, our pipeline requires significantly less manual annotation and has potential to grow beyond a relatively small number of POIs (i.e., celebrities) and to take advantage of mil-

*This material is based on work supported by NSF Cyberlearning grants 1822768 and 1551594.

lions of videos on YouTube, Vimeo, and other sites. In particular: (1) We detect potential overlap in the sets of speakers in multiple videos, rather than simply assuming they are disjoint based on a keyword search. This allows us to search using broader keywords (rather than for a single person) and potentially harvest more speakers. (2) We infer, using a combination of speaker and face embeddings, the number of speakers per video, which can give higher accuracy at selecting the relevant audio segments for each speaker.

3. PIPELINE TO COLLECT DISTINCT SPEAKERS

Our proposed procedure (see Fig. 1) to automatically collect a large number of audio clips that all have *distinct* speakers is based on two key ideas: (1) We can use an existing speaker embedding model (e.g., x-vector) to "bootstrap" the data collection if we impose a high confidence threshold to decide whether two randomly chosen videos are from the same speaker. (2) We can improve the accuracy in determining whether two videos contain the same speaker by using multimodal features such as face embeddings [7]. Like speaker embeddings, face embedding models are trained to map the input features (e.g., face pixels) into an embedding space that separates examples by the persons' identities.

Our procedure works at a high level as follows: (1): First, we scavenge a large number of videos from YouTube using a broad keyword search. (2): Next, we extract both speaker and face embeddings at many timesteps from every video. (3): Using these embeddings for each video as well as a simple linear classification model, we remove videos that are likely to contain multiple speakers. (4) and (5): Given a set of videos that all contain a single person speaking, we greedily add new videos to our collection $\mathcal{J}$ of videos, making sure that no video i that we might add contains *any* speaker in *any* video already added to $\mathcal{J}$. We describe the details below.

3.1. BookTube

As our source of audio clips, we retrieved URLs from YouTube using the keyword "BookTube". BookTube videos are where people share their thoughts and opinions on books. These are useful for our goal of collecting audio clips from independent speakers because (1) the face of the speaker is usually visible, and (2) each video usually contains just one person. In total we retrieved 38,707 unique URLs.

3.2. Embedding extraction

Speaker Embeddings (x-vector): We used the Kaldi speech processing toolkit [8] to extract speaker embeddings, in particular the pre-trained x-vector embedding model [3]. We used the WebRTC [9] and Librosa [10] Voice Activity Detectors to identify segments of the audio with presence of human speech (i.e., we required both detectors to output True). Speech segments were then split into 2-second consecutive parts, from which we extracted the speaker embeddings. **Face Embeddings (FaceNet)**: Using the MTCNN face detector [11], we automatically detected faces in each video frame of every video. Each face was then mapped into the embedding space using the pre-trained model from the Dlib library [12].

3.3. Removing Videos with Multiple Speakers

Some BookTube videos do contain multiple speakers. In stage 3 of our pipeline we remove videos with multiple speakers so as to avoid the issue of knowing who is speaking *when* (diarization) within each video. To automate this process, we trained a simple linear classifier that weights features computed from the embedding vectors. In particular, the feature set consists of the mean and standard deviation of the cosine similarities between all pairs of x-vectors, and all pairs of FaceNet embeddings, within the video. We also added one more feature, defined as $\frac{\text{SizeOfLargestCluster}}{\text{TotalNumberOfFaceNetEmbeddings}}$, based on the output of the Chinese Whisper clustering algorithm [13]. These features were then classified using logistic regression to predict whether the video contains 1 vs. > 1 speakers.

To train this model, we manually annotated 300 BookTube videos with whether or not they contained multiple speakers. (Note this only needs to be done once.) We then fit the logistic regressor on 225 labeled videos and tested on 75 videos. The model achieved 0.947 for Area Under the ROC (AUC) curve and 88% classification accuracy. Using this test set, we selected 0.855 as the threshold to keep the number of false positives at 0 on the test set (i.e., never classify a video with multiple speakers as "single speaker"). Finally, we re-trained the logistic regression model on all 300 videos.

3.4. Disjoint Speaker Sets

Given the collections of x-vectors and FaceNet embeddings for all the downloaded BookTube videos, we performed the following procedure to find a large subset of them such that no two videos contain the same speaker at any moment in time: We start by adding a random video from our entire BookTube dataset to the empty filtered collection $\mathcal{J}$. Subsequently, every time a new video was inspected, we compare the x-vectors and FaceNet embeddings of that video with those of all videos in $\mathcal{J}$. Let x_m^i and x_n^j represent the mth embedding vector from video i and the nth embedding vector from video j, respectively, where $m \in \{1, ..., M\}$ and $n \in \{1, .., N\}$. We estimate the probability that x_m^i and x_n^j were generated by the same speaker by the cosine similarity between them, i.e., $\cos(x_m^i, x_n^j)$. Then, to compute the probability that videos i and j contain speech from the same speaker at *any* moment, we aggregate over all $M \times N$ embedding vectors from the two videos with a function g (see details below) and then compare the result to a threshold τ. If the aggregate score $g(\{\cos(x_m^i, x_n^j)\}_{m,n}) < \tau$ for *both* the x-vectors *and* the

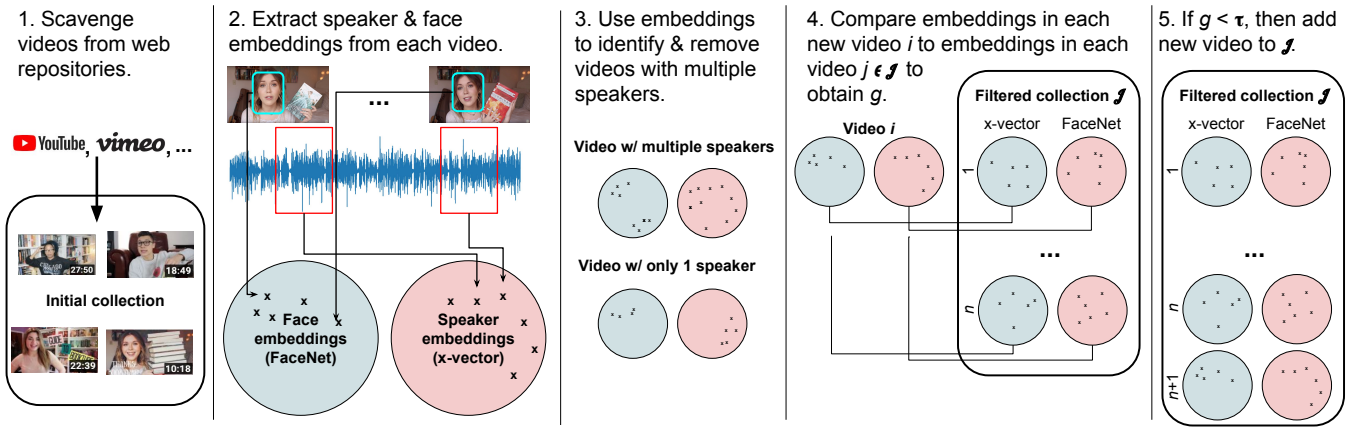

Fig. 1. Automated procedure to collect audio samples from many distinct speakers. The approach harvests samples from a *video* dataset (e.g., YouTube) and uses both face and speaker embeddings to "bootstrap" a large dataset.

FaceNet vectors, then videos i and j are deemed to contain speech from disjoint sets of speakers.

Choosing the aggregation function: Should a single, brief instant when two faces (or voices) contained in two videos looked/sounded very similar automatically "disqualify" both videos from being included in the dataset? If so, this implies that g should compute the *maximum* similarity. Or should only consistently similar values between the embeddings in two videos be considered from the same speaker? This suggests suggests g should compute the *average*. Based on some pilot experimentation, we used a hybrid approach: for face embeddings, $g_{\text{fa}} = \max$; for $g_{\text{sp}} =$ mean.

Choosing the thresholds: To decide the thresholds on the g functions, we manually labeled a subset of $n = 91$ BookTube videos (this need only be done once). For each of the $n(n-1)/2$ pairs (i, j), we labeled whether they contained the same speaker at any moment. Thresholds ($\tau_{\text{sp}} = 0.8, \tau_{\text{fa}} = 0.97$) on the two embeddings (speech, face) were computed so as to maximize the number of accepted videos while never accepting two videos as "distinct" if they actually contained the same speaker. From the 4095 pairs, 3947 contained distinct speakers and 148 contained the same speaker. This suggests that a random search through BookTube, while useful for finding videos with many distinct speakers, should still be filtered to remove videos with overlapping speaker sets.

3.5. Adding new videos to $\mathcal{J}$

From the downloaded videos that each contain only a single speaker, our goal is to find the largest possible subset such that no two videos contain overlapping sets of speakers. This is the *maximum independent set* problem, which is NP-hard. Fortunately, a simple greedy heuristic produces an independent set roughly half as large as the maximum independent set [14]. We can estimate how many videos must be downloaded to obtain the target number of *distinct* speaker recordings n in the following way: Consider a graph whose nodes consist of videos such that nodes i and j are adjacent iff their associated sets of speakers is overlapping. If the average graph degree is d and the number of vertices is V, then the expected size of the largest independent set (for a sparse random graph) obtained with a trivial greedy heuristic is $\frac{V}{d} \log d$ [14]. Hence, the size of the downloaded set of videos grows linearly in the desired number of videos with *distinct* speakers. The computational cost grows quadratically in V, but the task is parallelizable.

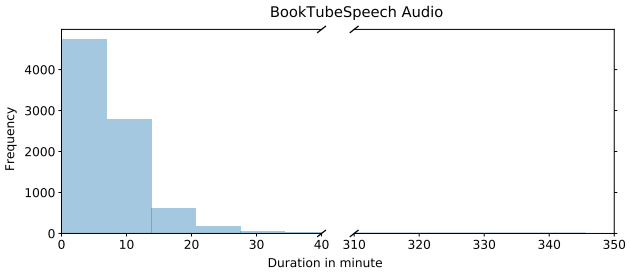

Fig. 2. Histogram of BookTubeSpeech audio duration.

4. NEW DATASET: BOOKTUBESPEECH

Based on the pipeline above, we pruned our initial collection of 38,707 BookTube videos down to a collection of 8,450 videos with distinct speakers. The BookTubeSpeech dataset is freely available for academic use[1]. The directory contains the extracted audio files (in `.wav` format) of all 8,450 videos. The average duration of all the files is 7.74 minutes. Most videos are less than 20 minutes, but 48 are longer than 40 min and one is about 350 min long. See histogram in Fig. 2.

In contrast to VoxCeleb, BookTubeSpeech contains one

[1] https://users.wpi.edu/~jrwhitehill/BookTubeSpeech.html

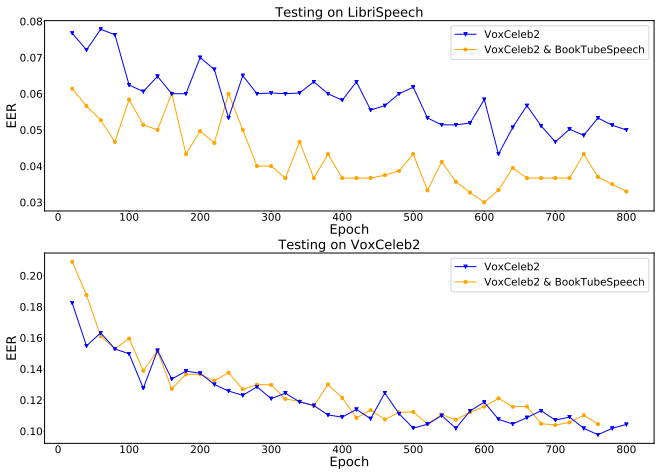

Fig. 3. EER of speaker verification models trained on VoxCeleb2 vs. VoxCeleb2+BookTubeSpeech, and tested on LibriSpeech (upper) or VoxCeleb2 (lower).

video per speaker. Typically, the recording conditions (e.g., microphone and its placement relative to the speaker) of BookTubeSpeech are the same throughout each video. The videos are long enough to span a variety of vocabulary and content, but each video is typically about one book. BookTubeSpeech can be used to train Universal Background Models [15] as well as speaker embedding models.

5. EXPERIMENTS

5.1. Multiple Embeddings

For the task of testing whether two videos contain overlapping sets of speakers, we compared the accuracy of FaceNet+(x-vector), to just x-vector and FaceNet embeddings by themselves. **Results**: On the 4,095 video pairs that were labeled manually from 91 BookTube videos, the AUC in distinguishing *disjoint speakers* from *overlapping speakers* was 0.982 for FaceNet vectors versus 0.968 for x-vector, suggesting that speech embeddings are somewhat more informative (at least for BookTube videos). By using both these representations in a simple linear classifier (optimal weights were 0.8 for FaceNet and 0.2 for x-vector), we can raise the AUC slightly to 0.984. While the difference (0.002) is small in absolute terms, it is a 10% relative reduction in classification error.

5.2. Speaker Verification

A strong test of a speaker embedding model, and the dataset used to train it, is to what extent it can produce a useful speaker verification system. In particular, we assessed whether combining BookTubeSpeech with VoxCeleb2 can yield a more accurate speaker embedding than training on VoxCeleb2 by itself. We measured test accuracy on the 118 test videos from VoxCeleb2 as described in [4], and also on 100 distinct speakers from LibriSpeech's "clean" data subset with a 50-50 female-male split. BookTubeSpeech is more similar to LibriSpeech in terms of speaking and recording conditions: These datasets have little to no background noise, and the microphone is positioned directly in front of the speaker. In contrast, VoxCeleb2 tends to be noisier and the microphone position varies.

Architecture: We trained a speaker verification model using MFCC features (window size of 0.025s, window step of 0.01s, and 40 filter-band banks). We used the same network and training method as Li Wan, etc. [16]. Inputs to the model have a fixed length of 160 frames.

Training: Each minibatch contains 4 different speakers, and each speaker has 6 randomly selected utterance samples (3 for enrollment, 3 for testing). We train each model for 800 epochs using Adam.

Testing: During testing, every minibatch contains 10 different speakers and each speaker has 6 random selected utterance samples. We iterate over all speakers in the test set 50 times, and we fix the random seed at the beginning of testing so that the results across methods are comparable.

Results: Fig. 3 shows the Equal Error Rate (EER) of speaker verification models trained on either VoxCeleb2 or on VoxCeleb2+BookTubeSpeech. When tested on LibriSpeech, the addition of the BookTubeSpeech training data substantially improved the EER across almost the entire epoch range, with a relative error reduction of around 30%. This provides evidence that BookTubeSpeech is useful for training speaker embeddings. On VoxCeleb2, adding the BookTubeSpeech training data made little difference. This is not too surprising, as VoxCeleb2 and BookTubeSpeech are quite different in their recording and speaking conditions. We are currently exploring different keyword searches (e.g., "travel vlog") for which we can collect new data using the pipeline in Fig. 1 that exhibit greater variability in these conditions.

6. CONCLUSIONS

We have presented and released for academic use a new speech dataset, BookTubeSpeech, which can be used to train new speaker embedding models. We also presented a novel data collection pipeline that uses both speaker and face embeddings to remove videos with overlapping sets of speakers. Importantly, our algorithm requires no manual annotation or pre-selection of Persons of Interest and thus can scale to generate much larger speaker datasets with minimal human labor. Experiments on speaker embeddings models show that BookTubeSpeech can be used to improve the accuracy of speaker verification models on LibriSpeech. **Future work** will assess how the label accuracy of speech datasets such as VoxCeleb2 and BookTubeSpeech influences the accuracy of downstream embedding model trained from these datasets.

7. REFERENCES

[1] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[2] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4052–4056.

[3] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[4] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "Voxceleb2: Deep speaker recognition," 09 2018, pp. 1086–1090.

[5] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "Voxceleb: A large-scale speaker identification dataset," 06 2017.

[6] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[7] Florian Schroff, Dmitry Kalenichenko, and James Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[8] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukáš Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlíček, Yanmin Qian, Petr Schwarz, Jan Silovský, Georg Stemmer, and Karel Vesel, "The kaldi speech recognition toolkit," *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 01 2011.

[9] John Wiseman, "Python interface to the webrtc voice activity detector," `https://github.com/wiseman/py-webrtcvad`, 2016.

[10] Brian McFee, Colin Raffel, Dawen Liang, Daniel Patrick Whittlesey Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto, "librosa: Audio and music signal analysis in python," 2015.

[11] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, Oct 2016.

[12] Davis E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[13] Chris Biemann, "Chinese whispers: An efficient graph clustering algorithm and its application to natural language processing problems," *Proceedings of TextGraphs*, pp. 73–80, 07 2006.

[14] Geoffrey R Grimmett and Colin JH McDiarmid, "On colouring random graphs," in *Mathematical Proceedings of the Cambridge Philosophical Society*. Cambridge University Press, 1975, vol. 77, pp. 313–324.

[15] Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[16] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.